A User-Customizable Xwindow-like GUI System for Mobile Phones

Jianming Wu          Toshiaki Uemukai          Fumiaki Sugaya
KDDI R&D Laboratories Inc., Japan
{ji-wu, to-uemukai, fsugaya}@kddilabs.jp

1 Introduction
In this paper, we propose a user-customizable Xwindow-like GUI system for mobile phones, which utilizes external devices to provide a suitable UI environment for a user. The proposed system can run multiple applications simultaneously on a mobile phone just like a personal computer and collaborate with different types of external devices such as mouse, keyboard, acceleration sensor, display, and so forth.

2 Xwindow-like GUI system for mobile phones
Due to the limitation of hardware performance, the mobile phone kernel uses RTOS (real time operating system) whereby an application runs with a single thread and a single window. In Japan, 3G services using the application platform BREW (BREW) have been advanced. And by using the BREW development platform, we can develop advanced applications running on mobile phones. However, the BREW platform does not support multi-threads yet.

Therefore, we developed an Xwindow-like GUI system for mobile phones with BREW, called “K-tai win.” K-tai win supports multi-deskops, multi-windows, and time-sharing multi-threads that support concurrency process. Multiple applications, each of which has one or more windows, can run on a desktop, and multiple desktops can be switched. Moreover, since we developed K-tai win as an application development platform, a developer can develop their own BREW application with Xwindow-like GUI easily. The platform supports C/C++ and an API extension of BREW. Therefore, users can operate a user-friendly interface on a mobile phone more conveniently.

Figure 1 shows the prototype system of K-tai win. This is developed by using K-tai win’s API to create the rich window GUI objects of a button, listbox, and tab.

3 User-customizable GUI
Traditionally, since the main input interface of a mobile phone is a ten-key pad, users’ operations are limited. T9 predicts what a user wants to type on a mobile phone so that he/she may push a key just once per letter [T9]. On the other hand, recently Bluetooth technology has been developed, which can connect to a mobile phone via a short-range radio frequency and exchange information between them. Therefore, considering the extensibility, generality, and usability, it may be effective that a mobile phone collaborates with external devices via a wireless communication technology such as Bluetooth. We propose some applications whereby K-tai win collaborates with external devices as follows:

(1) Operation with a mouse
Figure 2 shows an application of the operation with a Bluetooth mouse for K-tai win. When the connection between the mouse and the mobile phone is established, the click and the move event of the mouse will be transferred to the mobile phone, so that a user can operate windows more freely.

(2) Operation with an acceleration sensor
Figure 3 shows an application of the operation with an acceleration sensor. When the sensor is placed on a mobile phone and a user rolls the mobile phone with his/her hand, the sensor can measure its acceleration and tilt. Here according to the direction and amount of tilt, this application switches the showing of a keyboard. When the user shakes the mobile phone, this application switches the mode to input characters, e.g. large or small letters.

(3) Collaboration with an external display
Figure 4 shows an application of the collaboration with an external display. K-tai win can manage multiple windows and desktops simultaneously. Then, when a user makes a presentation, two desktops (or windows) are displayed on a large display and a screen respectively and simultaneously. A user can also control them by operating other desktops (or windows) displayed on the mobile phone.

As mentioned above, K-tai win can collaborate with many kinds of external devices. With K-tai win, a user can also use his/her favorite device to operate an application as he/she wishes and associate his/her own actions with its behavior. For example, with a mouse or tracking ball, the user can associate its wheel or ball event with such behavior to change the size or the position of a window. With an acceleration sensor, the user can associate its tilt event with such behavior to change or scroll a window. It may be quite convenient to scroll the screen when the user browses a large map and a web page which is conventionally browsed on a PC. Moreover, the user can also associate a shaking event with the “double click” behavior. Furthermore, with a GPS sensor, we can store such customized operation information with the location data, so that K-tai win can switch to the most convenient desktop environment automatically when we move from place to place.

4 Conclusion
We developed a user-customizable Xwindow-like GUI system for mobile phones. Regardless of the restriction of its interface, this system has the ability to support infinite customization of its uses. In the next stage, we are planning to find other external devices which may be able to collaborate with this system, and develop applications and effective APIs.

Reference
BREW. http://brew.qualcomm.com/brew/.